BBN:
Description of the PLUM System as Used for MUC-5

*The PLUM System Group**

BBN Systems and Technologies
70 Fawcett Street
Cambridge, MA 02138
weischedel@bbn.com

APPROACH

Traditional approaches to the problem of extracting data from texts have emphasized hand-crafted linguistic knowledge. In contrast, BBN's **PLUM** system (**P**robabilistic **L**anguage **U**nderstanding **M**odel) was developed as part of an ARPA-funded research effort on integrating probabilistic language models with more traditional linguistic techniques. Our research and development goals are:

- more rapid development of new applications,
- the ability to train (and re-train) systems based on user markings of correct and incorrect output,
- more accurate selection among interpretations when more than one is found, and
- more robust partial interpretation when no complete interpretation can be found.

We began this research agenda approximately three years ago. During the past two years, we have evaluated much of our effort in porting our data extraction system (PLUM) to a new language (Japanese) and to two new domains.

KEY SYSTEM FEATURES

Three key design features distinguish PLUM: statistical language modeling, learning algorithms and partial understanding. The first key feature is the use of *statistical modeling to guide processing*. For the version of PLUM used in MUC-5, part of speech information was determined by using well-known Markov modeling techniques embodied in BBN's part-of-speech tagger POST [5]. We also used a correction model, AMED [3], for improving Japanese segmentation and part-of-speech tags assigned by JUMAN. For the microelectronics domain, we used a probabilistic model to help identify the role of a company in a capability (whether it is a developer, user, etc.). Statistical modeling in PLUM contributes to portability, robustness, and trainability.

The second key feature is our use of *learning algorithms* both to obtain the knowledge bases used by PLUM's processing modules and to train the probabilistic algorithms. We feel the key to portability of a data extraction system is automating the acquisition of the knowledge bases that need to change for a particular language or application. For the MUC-5 applications we used learning algorithms to train POST, AMED, and the template-filler model mentioned above. We also used a statistical learning algorithm to learn case frames for verbs from examples (the algorithm and empirical results are in [4]).

A third key feture is *partial understanding*, by which we mean that all components of PLUM are designed to operate on partially interpretable input, taking advantage of information when available, and not failing when information is unavailable. Neither a complete grammatical analysis nor complete semantic interpretation is required. The system finds the parts of the text it can understand and pieces together a model of the whole from those part and their context.

PROCESSING STAGES

The PLUM architecture is presented in Figure 1. Ovals represent declarative knowledge bases; rectangles represent processing modules. A more detailed description of the system components, their individual outputs, and their knowledge bases is presented in Ayuso et al., [1]. The processing modules are briefly described below.

---

* Ralph Weischedel (Principal Investigator), Damaris Ayuso, Sean Boisen, Heidi Fox, Robert Ingria, Tomoyoshi Matsukawa, Constantine Papageorgiou (BBN), Dawn MacLaughlin, Masaichiro Kitagawa, Tsutomu Sakai (Boston University), June Abe, Hiroto Hosihi, Yoichi Miyamoto (University of Connecticut), and Scott Miller (Northeastern University)

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
1993		**00-00-1993 to 00-00-1993**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
BBN: Desccription of the PLUM System as Used for MUC-5	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **BBN Technologies,10 Moulton Street,Cambridge,MA,02238**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**		**15**	

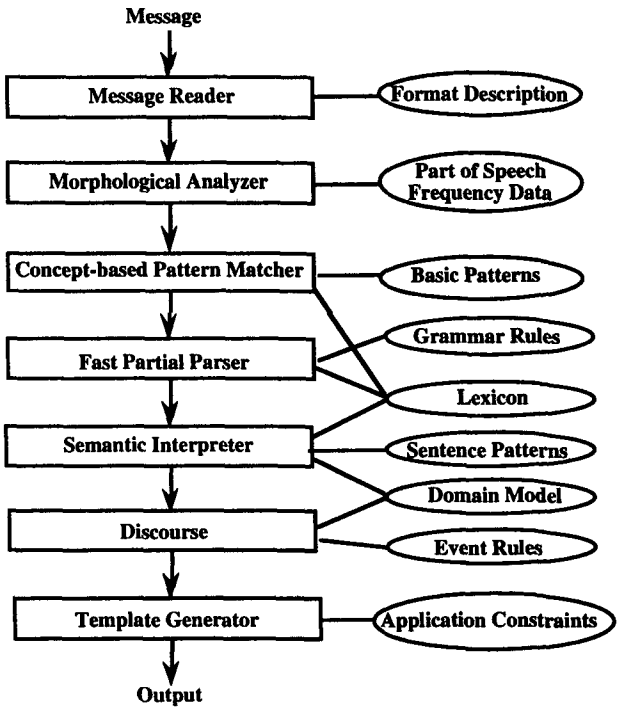

Figure 1: PLUM System Architecture: Rectangles represent domain-independent, language-independent algorithms; ovals represent knowledge bases.

Message Reader

This module is like the "text zoner" of Hobbs' description of generic data extration systems. PLUM's specification of the input format is a declarative component of the message reader, allowing the system to be easily adapted to handle different formats. The input to the PLUM system is a file containing one or more messages. The message reader module determines message boundaries, identifies the message header information, and determines paragraph and sentence boundaries. To date, we have designed format specifications for about half a dozen domains.

Morphological Analyzer

The first phase of processing is assignment of part-of-speech information, e.g., proper noun, verb, adjective, etc. In BBN's part-of-speech tagger POST [5], a bi-gram probability model, frequency models for known words (derived from large corpora), and probabilities based on word endings for unknown words are employed to assign part of speech to the highly ambiguous words and unknown words of the corpus. POST tags each word with one of 47 possible tags with 97% accuracy for known words. For the Japanese domains, JUMAN is used to propose word segmentation and part-of-speech assignments, which are then corrected by AMED [3] before being handed to POST for final disambiguation. Below are the part-of-speech tags produced by POST for the first sentence of the EJV walkthrough article 0592:

"BRIDGESTONE SPORTS CO. SAID FRIDAY IT HAS SET UP A JOINT VENTURE IN TAIWAN WITH A LOCAL CONCERN AND A JAPANESE TRADING HOUSE TO PRODUCE GOLF CLUBS TO BE SHIPPED TO JAPAN."

(BRIDGESTONE NP) (SPORTS NPS) (CO. NP) (SAID VBD) (FRIDAY NP) (IT PP) (HAS VBZ)
(SET UP VBN) (A DT) (JOINT VENTURE NN) (IN IN) (TAIWAN NP) (WITH IN) (A DT) (LOCAL JJ) (CONCERN NN)
(AND CC) (A DT) (JAPANESE JJ) (TRADING HOUSE NN) (TO TO) (PRODUCE VB) (GOLF NN) (CLUBS NNS) (TO TO)
(BE VB) (SHIPPED VBN) (TO TO) (JAPAN NP) (. .)

Concept-Based Pattern Matcher

The Concept-based Pattern Matcher was developed after MUC-4 to deal with grammatical forms, such as corporation names. It applies finite state patterns to the input, which consists of word tokens with part-of-speech and semantic concept information. In particular, word groups that are important to the domain and that may be detectable with only local syntactic analysis can be treated here. When a pattern is matched, a semantic form is

assigned by the pattern. In both joint ventures and microelectronics, patterns were used to group proper nouns into company names, organization names, and person names. Continuing with the example sentence discussed above, a pattern recognized the sequence (BRIDGESTONE NP) (SPORTS NPS) (CO. NP) as a company; the pattern's action substituted the single token (BRIDGESTONE SPORTS CO. CORP), with semantics of corporation.

Fast Partial Parser (FPP)

The FPP is a near-deterministic parser which generates one or more non-overlapping parse fragments spanning the input sentence, deferring any difficult decisions on attachment ambiguities. When cases of permanent, predictable ambiguity arise, the parser finishes the analysis of the current phrase, and begins the analysis of a new phrase. Therefore, the entities mentioned and some relations between them are processed in every sentence, whether syntactically ill-formed, complex, novel, or straightforward. Furthermore, this parsing is done using essentially domain-independent syntactic information.

FPP averages about 6 fragments for sentences as complex as in the EJV corpus; this number is inflated since punctuation usually results in an isolated fragment. Continuing with the same example sentence, Figure 2 shows nine parse fragments as generated by FPP. The Japanese grammar produces smaller fragments by design.

Semantic Interpreter

The semantic interpreter contains two sub-components: a rule-based fragment interpreter and a pattern-based sentence interpreter. The first was used in MUC-3 and MUC-4. The rule-based fragment interpreter applies semantic rules to each fragment produced by FPP in a bottom-up, compositional fashion. Semantic rules are matched based on general syntactic patterns, using wildcards and similar mechanisms to provide robustness. A semantic rule creates a semantic representation of the phrase as an annotation on the syntactic parse. A semantic formula includes a variable (e.g., ?13), its type, and a collection of predicates pertaining to that variable. There are three basic types of semantic forms: entities in the domain, events, and states of affairs. Each of these can be further categorized as known, unknown, and referential. Entities correspond to the people, places, things, and time intervals of the domain. These are related in various ways, such as through events (who did what to whom) and states of affairs (properties of the entities). Entity descriptions typically arise from noun phrases; events and states of affairs are often described in clauses.

The rule-based fragment interpreter encodes defaults so that missing semantic information does not produce errors, but simply marks elements or relationships as unknown. Partial understanding is critical to text processing systems; missing data is normal. For example, the generic predicate PP-MODIFIER indicates that two entities are connected via a certain preposition. In this way, the system has a "placeholder" for the information that a certain structural relation holds, even though it does not know what the actual semantic relation is. Sometimes understanding the relation more fully is of no consequence, since the information does not contribute to the template-filling task. The information is maintained, however, so that later expectation-driven processing can use it if necessary.

```
F1: "BRIDGESTONE SPORTS CO. SAID FRIDAY IT HAS
SET UP A JOINT VENTURE"
  (S (NP (N (NAME "BRIDGESTONE SPORTS CO.")))
    (VP (AUX)
      (VP (V "SAID")
        (NP (MONTH "FRIDAY"))
        (S
          (S (NP (PRO-DET-SPEC "IT"))
            (VP (AUX (V "HAS"))
              (VP (V "SET UP")
                (NP (DETERMINER "A")
                    (N "JOINT VENTURE")))))))))
F2: "IN TAIWAN"
  (PP (PREP "IN")
      (NP (N (NAME "TAIWAN"))))
F3: "WITH A LOCAL CONCERN"
  (PP (PREP "WITH")
      (NP (DETERMINER "A")
          (ADJP (ADJ "LOCAL"))
          (N "CONCERN")))
F4: "AND"
  (CONJ "AND")
F5: "A JAPANESE TRADING HOUSE"
  (NP (DETERMINER "A")
      (ADJP (ADJ "JAPANESE"))
      (N "TRADING HOUSE"))
F6: "TO PRODUCE GOLF CLUBS"
  (VP (AUX (TO "TO"))
      (VP (V "PRODUCE")
          (NP (N "GOLF") (N "CLUBS"))))
F7: "TO"
  (PREP "TO")
F8: "BE SHIPPED TO JAPAN"
  (VP (AUX (V "BE"))
      (VP (V "SHIPPED")
          (PP (PREP "TO")
              (NP (N (NAME "JAPAN"))))))
F9: "."
  (PUNCT ".")
```

Figure 2. Parser Output: Partial parse found for the example sentence.

An important consequence of the fragmentation produced by FPP is that top-level constituents are typically more shallow and less varied than full sentence parses. As a result, a fairly high level of semantics coverage can be obtained quite quickly when the system is moved to a new domain. This would not be possible if the semantic rules were required to cover a wider variety of syntactic structures before it could achieve reasonable performance. In this way, semantic coverage can be added gradually, while the rest of the system is progressing in parallel.

The second sub-component of the semantic interpreter module is a pattern-based sentence interpreter which applies semantic pattern-action rules to the semantics of each fragment of the sentence. This replaces the fragment combining component used in MUC-4. The semantic pattern matching component employs the same core engine as the concept-based pattern matcher. These semantic rules can add additional long-distance relations between semantic entities in different fragments within a sentence. For example, in the English joint-venture domain, we have defined a rule which looks for a sequence of [<ENTITY> "capitalized at" <MONETARY-AMOUNT>]. This rule's action creates an OWNERSHIP semantic form, where <ENTITY> is related via the OWNERSHIP-OWNED role and <MONETARY-AMOUNT> via the OWNERSHIP-CAPITALIZATION role.

The semantic lexicon is separate from the parser's lexicon and has much less coverage. Lexical semantic entries indicate the word's semantic type (a domain model concept), as well as predicates pertaining to it. For example, here is the lexical semantics for the noun collocation "joint venture". This entry indicates that the semantic type is JOINT-VENTURE, and that a "with" or "between" PP argument whose type is ENTITY should be given the role PARENT-OF, and a "for" PP argument of type ACTIVITY should be given the role ACTIVITY-OF.

```
(defnoun "joint venture"
        (JOINT-VENTURE (:CASE (("with" "between") ENTITY PARENT-OF) ("for" ACTIVITY ACTIVITY-OF))))
```

We used an automatic case frame induction procedure to construct an initial version of the lexicon [4]. Word senses in the semantic lexicon have probability assignments. For MUC-5 probabilities were (automatically) assigned so that each word sense is more probable than the next sense, as entered in the lexicon.

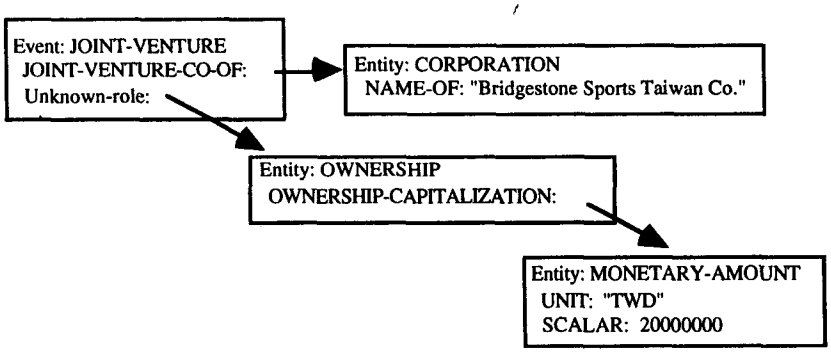

Figure 3. Semantic Structure: The semantic representation for the first fragment in Figure 2.

In Figure 3 we show the semantic representation that is built for the phrase "THE JOINT VENTURE, BRIDGESTONE SPORTS TAIWAN CO., CAPITALIZED AT 20 MILLION NEW TAIWAN DOLLARS" in EJV walkthrough article 0592 (this phrase is parsed within a single fragment by FPP). Notice that the JOINT-VENTURE is linked to the OWNERSHIP information via an unknown role, because the interpreter was unable to determine a specific relationship between the NP "THE JOINT VENTURE, BRIDGESTONE SPORTS TAIWAN CO.," and the participial modifier "CAPITALIZED AT ..." The discourse component will further refine the relationship between these two semantic objects to the JV-OWNERSHIP-OF relation.

Discourse Processing

PLUM's discourse component [2] performs the operations necessary to create a meaning for the whole message from the meaning of each sentence. The message level representation is a list of discourse domain objects (DDOs) for the top-level events of interest in the message (e.g., JOINT-VENTURE events in the joint-venture domain or CAPABILITY events in the microelectronics domain). The semantic representation of a phrase in the text only includes information contained nearby in a sentence; in creating a DDO, the discourse module must infer other long-distance or indirect relations not explicitly found by the semantic interpreter, and resolve any references in the text.

The discourse component creates two primary structures: a discourse predicate database and the DDOs. The database contains all the predicates mentioned in the semantic representation of the message. When references are resolved, corresponding semantic variables are unified. Any other inferences are also added to the database.

To create the DDOs, the discourse component processes each semantic form produced by the interpreter, adding its information to the database and performing reference resolution for pronouns and anaphoric definite NPs. Set- and member-type references may be treated. When a semantic form for an event of interest is encountered, a DDO is generated, and any slots already found by the interpreter are filled in. The discourse processor then tries to merge the new DDO with a previous DDO, in order to account for the possibility that the new DDO might be a repeated reference to an earlier one.

Once all the semantic forms have been processed, heuristic rules are applied to fill any empty slots by looking at the text surrounding the forms that triggered a given DDO. Each filler found in the text is assigned a confidence score based on distance from trigger. Fillers found nearby are of high confidence, while those farther away receive worse scores (low numbers represent high confidence; high numbers low confidence; thus 0 is the "highest" confidence score).

Following is the DDO for the first JOINT-VENTURE in EJV walkthrough article 0592:

```
DDO: JOINT-VENTURE
Trigger fragments:
  "BRIDGESTONE SPORTS CO. SAID FRIDAY IT HAS SET UP A JOINT VENTURE"
  "THE JOINT VENTURE, BRIDGESTONE SPORTS TAIWAN CO., CAPITALIZED AT 20 MILLION
     NEW TAIWAN DOLLARS, WILL START PRODUCTION IN JANUARY 1990"
-------------------------------------------------------------------------------------------------------------------
  JOINT-VENTURE-CO-OF:        "BRIDGESTONE SPORTS TAIWAN CO." (score = 0)
  JV-PARENT-OF:               "BRIDGESTONE SPORTS CO." (score = 1)
                              "A LOCAL CONCERN" (score = 2)
                              "A JAPANESE TRADING HOUSE" (score = 2)
                              "GOLF CLUBS" (score = 2)
                              "CLUBS" (score = 2)
  JV-ACTIVITY-OF:             "start production" (score = 1)
                              "produce golf clubs" (score = 2)
                              "be shipped to Japan" (score = 2)
                              "with production of 20,000 iron" (score = 2)
  JV-OWNERSHIP-OF:  "capitalized at 20 million new Taiwan dollars" (score =1)
```

Each trigger fragment contains one or more words whose semantics triggered this DDO. A DDO can have multiple trigger fragments if the discourse component determines that the triggers corefer. In this example, a "joint venture" in the first fragment co-refers with "the joint venture" in the second fragment. A score of 0 indicates the filler was found directly by the semantics; 1 that it was found in the same fragment as a trigger form; and 2 in the same sentence.

Template Generation

The template generator takes the DDOs produced by discourse processing and fills out the application-specific templates. Clearly, much of this process is governed by the specific requirements of the application, considerations which have little to do with linguistic processing. The template generator must address any arbitrary constraints, as well as deal with the basic details of formatting.

The template generator uses a combination of data-driven and expectation-driven strategies. First the DDOs found by the discourse module are used to produce template objects. Next, the slots in those objects are filled using information in the DDO, the discourse predicate database, or other sources of information such as the message header (e.g., document number, document source, and date information), statistical models of slot filling (e.g., as in the microelectronics domain to choose among the slots: purchaser/user, developer, distributor, and manufacturer), or from heuristics (e.g., the status of an equipment object is most likely to be IN_USE, or the status of a joint venture object is most likely to be EXISTING).

Parameters in PLUM

Many aspects of PLUM's behavior can be controlled by simply varying the values of system parameters. For example, PLUM has parameters to control aspects of tagging, parsing, pattern matching, event merging and slot filling by discourse, and template filling. An important goal has been to make our system as "parameterizable" as possible, so that the same software can meet different demands for recall, precision, and overgeneration.

TRAINING DATA AND TECHNIQUES

The entire development corpus was used in various ways as training data. PLUM was run over all messages to detect, debug, and correct any causes of system breaks. The entity name slot for all messages was used to quickly add names to the domain-dependent lexicon. For both microelectronics applications, statistics on the co-occurrence of particular entities in various roles (developer, manufacturer, etc.) were used as a fall-back model for low-confidence relationships detected in the texts.

The TIPS1 and TIPS2 sets for all applications were used as blind test sets to measure our progress at least once a week. Throughout, we used the summary output from the scoring procedure to guide our development, rather than adding to the lexicon or debugging the system based on particular messages.

DEALING WITH MULTIPLE LANGUAGES AND MULTIPLE DOMAINS

Any system that participated in more than one domain in MUC-5 and/or in more than one language has demonstrated domain independence and language independence. In PLUM, the text zoner, morphological processing, parsing, and semantic interpretation employ language-independent and domain-independent algorithms driven by data (knowledge) bases. Similarly, the discourse algorithms and template generation algorithms are domain- and language-independent, and are driven by knowledge that is predominantly declarative.

The issue (or the goal) that all systems must address further is greater automation of the porting process. Our approach has been to rely on probabilistic learning algorithms. Based on our experience in the last two years, several conclusions have emerged:

1. **Porting PLUM to a new domain**, even in multiple languages, takes much less effort now. Table 1 shows the effort expended in porting PLUM to the microelectronics domain. In 52 person-days, PLUM was processing microelectronics articles in both English and Japanese, obtaining reasonable performance. Had we run PLUM at that time on the TIPS3 test sets, scores would already have been impressive in English (an ERR of 74). For Japanese, performance was 73 on test set TIPS2. (We quote the score for TIPS2, because it covered only the capabilities for which there was data at the time of the TIPS2 version of PLUM.)

Tasks	Person-Days
Language-independent	14
English	19
Japanese	19
TOTAL	52

Table 1: Effort to Port to Microelectronics

Language Domain Pair	Person-Months
English Joint Venture	4-5
Japanese Joint Venture	2
English Microelectronics	3
Japanese Microelectronics	3.5

Table 2: Total Effort in Each Domain

Table 2 lists our estimated effort in each domain; Figure 4 portrays the data graphically. The effort in each language was largely balanced; the performance of the system across languages and domains was also remarkably balanced, as shown graphically in Figure 5.

2. **Annotating data for PLUM's probabilistic model of a new language**, even with little language-specific resources, proved easier than anticipated. The only resource available to us at the start was the JUMAN system from Kyoto University, which hypothesizes word segmentation and part of speech for Japanese text.. Our Japanese speakers were able to annotate part of speech and word boundaries at about 1,000 words per hour, and were able to annotate syntactic structure at about 750 words per hour. Initial annotation and testing were performed using only 16,000 words plus the JUMAN lexicon; therefore, the initial port to Japanese required only about a person-week of annotation effort.

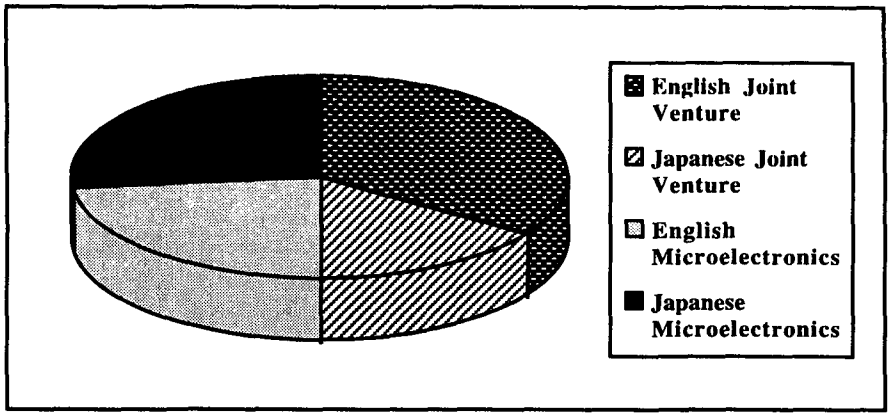

Figure 4: Distribution of Effort across Domains: Effort across languages was about equal.

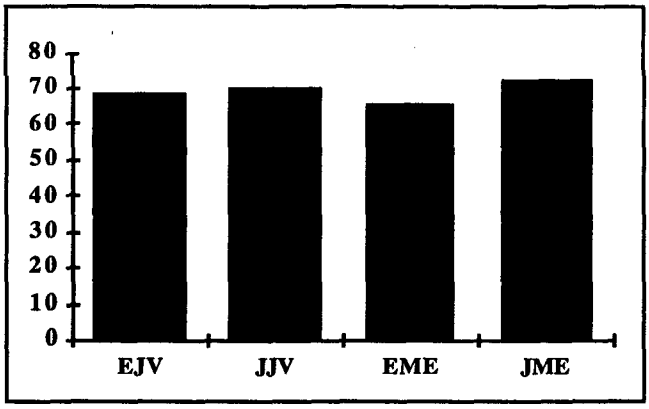

Figure 5: Performance Based on ERR: Across language-domain pairs, there was remarkable consistency in PLUM's performance.

3. **Building lexical resources** for a new language or a new domain took only a few person days using heuristics. In Japanese, a three step process for hypothesizing proper names reduced the labor involved. First, we ran JUMAN + POST over the training corpus to find the sequence of words and their most likely part of speech in context. Then, a finite-state process with a handful of language-specific patterns was run on the result to hypothesize (previously unknown) proper nouns in the corpus. The patterns were designed for high recall of names, at the expense of low precision; we measured the effectiveness of the technique as 90% recall at 20% precision. Lastly, a person ran through the hypothesized proper names using KWIC as a resource to quickly eliminate bad hypotheses. The resulting list of names was made available to all the participants in JJV.

A simple manual technique also enabled fast semantic categorization of the nouns and verbs of each domain in both languages. Using a KWIC index and the frequency of each noun and each verb in the corpus, we could define about 125 words per hour into categories such as HUMAN, CORPORATION, OFFICER, GOVERNMENT-ORGANIZATION, etc. The process could go so quickly by organizing the categories into small menus of at most 12 items, so that a person need only make simple discriminations in any pass through a list of words.

4. **Training new staff to use PLUM effectively** proved easier than anticipated. Our team faced training new staff two months before the MUC-5 test, as our single Japanese programmer needed to reduce his involvement substantially. Starting at the beginning of June, two Japanese computer science majors, who had just completed their junior year at college came to BBN. They had had no training in computational linguistics, but had had one course in artificial intelligence and one in LISP. In June, they learned about data extraction, the joint venture and microelectronics tasks, and how to use PLUM. Since the Japanese articles on packaging and lithography had arrived much later than the other data, and since we had not touched that data, they focussed on those two capabilities starting July 1. Initially, of course, PLUM had near 100 as an ERR on sets composed primarily of those

microelectronics capabilities. As evident in Figure 6, the progress was rapid and dramatic, as the error rate dropped by 25% in all cases and by almost 50% in some cases.

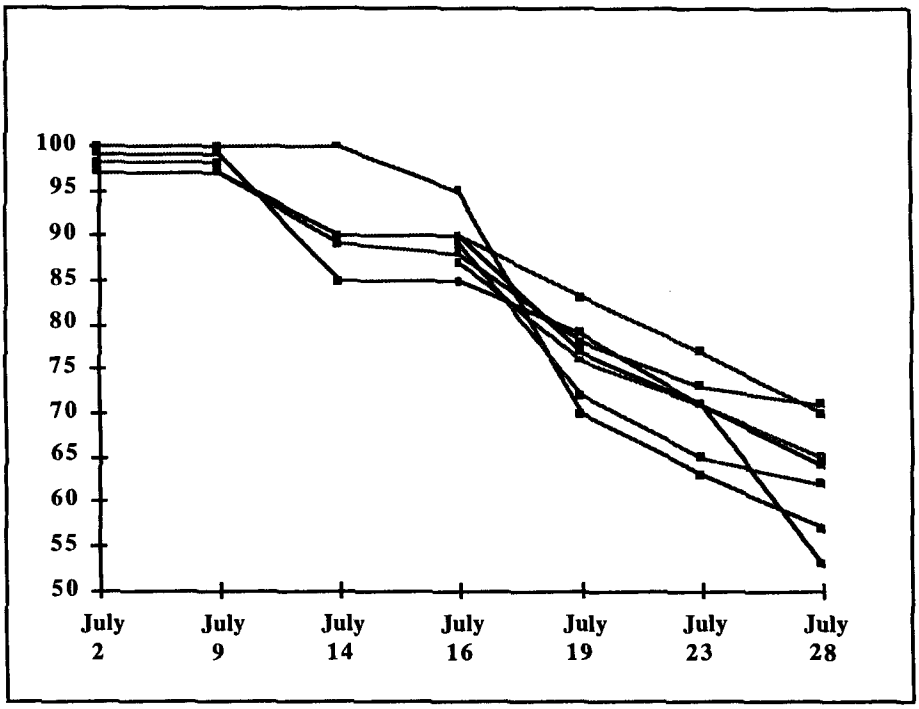

Figure 6: Progress in JME: For development messages involving packaging and lithography, progress of new staff with minimal training was rapid and dramatic.

CONCLUSIONS

We began our research agenda approximately three years ago when we build PLUM for MUC-3. During the past two years, we have focused much of our effort on techniques to facilitate porting our data extraction system (PLUM) to new languages (Japanese) and to two new domains (joint ventures and microelectronics), as well as infrastructure development.

Some of the lessons we learned during our work include the following:

- Automatic training and acquisition of knowledge bases can yield relatively good performance at reduced labor, as evidenced, for example, by a quick port to the microelectronics domain (in 2 languages) in 2 person-months (after which further refinements were made).
- Domains dominated by jargon (sub-language) may be easier than domains of normal vocabulary because there is less ambiguity and more predictability. For TIPSTER this means that the microelectronics domain was easier than joint ventures.
- Japanese was easier to process than English because of strong clues provided by case-markers, and a less varied linguistic structure in the articles.
- Availability of a large text corpus was invaluable for quick knowledge acquisition. A smaller number of filled templates should still be adequate.
- Our algorithms were already largely language- and domain-independent; an important goal remains to further automate the porting process.
- Finite-state pattern matching is a useful complement to linguistic processing, offering a good fall-back strategy for addressing language constructions that are hard to treat via general linguistically-based approaches.
- Continued work on discourse processing is important to improving performance. Reliably determining when different descriptions of events or objects in fact refer to the same thing remains one of the hardest problems in data extraction.
- Improving syntactic coverage is a priority. Increased coverage normally leads to greater perceived ambiguity in the system; we hope to counter this through the use of probabilistic models.

We plan to continue our research agenda emphasizing the use of probabilistic modeling and learning algorithms for data extraction in order to continue improving robustness and portability.

SYSTEM WALKTHROUGHS

No development was done on the walkthrough messages for any of the domains, prior to the MUC-5 test.

EJV Walkthrough (Message 0592)

Questions to Address:
(1) Coreference determination:
• *Which coreferences did your system get?*
[a]: *"LOCAL CONCERN",*
"UNION PRECISION CASTING CO. OF TAIWAN"

PLUM did not find this coreference. The system misanalyzed "Union Precision Casting Co." such that the name was split across 2 fragments.

System Response Template for 0592:

<TEMPLATE-0592-1> :=
DOC NR: 0592
DOC DATE: 241189
DOCUMENT SOURCE: "Jiji Press Ltd."
CONTENT: <TIE_UP_RELATIONSHIP-0592-1>
<TIE_UP_RELATIONSHIP-0592-2>
<TIE_UP_RELATIONSHIP-0592-3>
<TIE_UP_RELATIONSHIP-0592-1> :=
TIE-UP STATUS: EXISTING
ENTITY: <ENTITY-0592-1>
<ENTITY-0592-2>
JOINT VENTURE CO: <ENTITY-0592-2>
OWNERSHIP: <OWNERSHIP-0592-1>
ACTIVITY: <ACTIVITY-0592-1>
<TIE_UP_RELATIONSHIP-0592-2> :=
TIE-UP STATUS: EXISTING
JOINT VENTURE CO: <ENTITY-0592-3>
OWNERSHIP: <OWNERSHIP-0592-1>
ACTIVITY: <ACTIVITY-0592-1>
<ACTIVITY-0592-2>
<TIE_UP_RELATIONSHIP-0592-3> :=
TIE-UP STATUS: EXISTING
ENTITY: <ENTITY-0592-1>
<ENTITY-0592-2>
<ENTITY-0592-4>
OWNERSHIP: <OWNERSHIP-0592-1>
<ENTITY-0592-1> :=
NAME: BRIDGESTONE SPORTS CO
ALIASES: "BRIDGESTONE SPORTS"
TYPE: COMPANY
ENTITY RELATIONSHIP:
<ENTITY_RELATIONSHIP-0592-1>

<ENTITY_RELATIONSHIP-0592-3>
<ENTITY-0592-2> :=
NAME: TAIWAN
TYPE: COMPANY
NATIONALITY: JAPAN (COUNTRY)
ENTITY RELATIONSHIP:
<ENTITY_RELATIONSHIP-0592-1>
<ENTITY_RELATIONSHIP-0592-3>
<ENTITY-0592-3> :=
NAME: BRIDGESTONE SPORTS TAIWAN CO
TYPE: COMPANY
ENTITY RELATIONSHIP:
<ENTITY_RELATIONSHIP-0592-2>
<ENTITY-0592-4> :=
NAME: TAGA CO
TYPE: COMPANY
ENTITY RELATIONSHIP:
<ENTITY_RELATIONSHIP-0592-3>
<ENTITY_RELATIONSHIP-0592-1> :=
ENTITY1: <ENTITY-0592-1>
<ENTITY-0592-2>
ENTITY2: <ENTITY-0592-2>
REL OF ENTITY2 TO ENTITY1: CHILD
STATUS: CURRENT
<ENTITY_RELATIONSHIP-0592-2> :=
ENTITY2: <ENTITY-0592-3>
REL OF ENTITY2 TO ENTITY1: CHILD
STATUS: CURRENT
<ENTITY_RELATIONSHIP-0592-3> :=
ENTITY1: <ENTITY-0592-1>
<ENTITY-0592-2>
<ENTITY-0592-4>
REL OF ENTITY2 TO ENTITY1: CHILD
STATUS: CURRENT
<ACTIVITY-0592-1> :=
INDUSTRY: <INDUSTRY-0592-1>
<ACTIVITY-0592-2> :=
INDUSTRY: <INDUSTRY-0592-2>
<INDUSTRY-0592-1> :=
INDUSTRY-TYPE: PRODUCTION
PRODUCT/SERVICE: (- "GOLF CLUBS")
<INDUSTRY-0592-2> :=
INDUSTRY-TYPE: PRODUCTION
PRODUCT/SERVICE: (- "20,000 IRON")
<OWNERSHIP-0592-1> :=
OWNED: <ENTITY-0592-2>
OWNERSHIP-%: (<ENTITY-0592-1> 75)
(<ENTITY-0592-1> 15)

[b]: *"A JAPANESE TRADING HOUSE",*
"TAGA CO., A COMPANY ACTIVE IN TRADING WITH TAIWAN"

[c]: *"A JOINT VENTURE",*
"THE JOINT VENTURE, BRIDGESTONE SPORTS TAIWAN CO.",
"THE NEW COMPANY, BASED IN KAOHSIUNG, SOUTHERN TAIWAN",
"THE TAIWAN UNIT"

PLUM did not find any of these coreference relations.

[d]: *"BRIDGESTONE SPORTS CO.",*
"BRIDGESTON SPORTS",
"BRIDGESTONE SPORTS",
"THE JAPANESE SPORTS GOODS MAKER"

PLUM found "Bridgestone Sports Co" and "Bridgestone Sports" as being coreferential. PLUM did not recognize the "typo" alias, or the Japanese sport goods maker coreferences.

• *Of those, which could it have gotten 6 months ago (at the previous evaluation)?*

[a] & [b]: PLUM could not recognize these 6 months ago.

[c]: 6 months ago, PLUM did find coreference between "a joint venture" and "the joint venture".

[d]: PLUM could not get any of these 6 months ago.

• *How can you improve the system to get the rest?*

[a]: The phrase "local concern" is assigned a semantic type that is a superconcept of CORPORATION. If the discourse module allowed merging of a subconcept event into a superconcept event (something which is allowed in the microelectronics domain but not currently in the joint venture domain), then PLUM could potentially find this coreference via discourse event merging. However, PLUM's company name recognizer would need to be adapted so that it would not misanalyze the company name "Union Precision Casting Co."

[b]: This is a harder case. In order to find this coreference, PLUM would probably need to recognize that both mentions are involved in trading.

[c]: A more explicit treatment of definite references would help with these cases. Also, better recognition of locations would aid in establishing coreference between the two mentions of the Taiwan company.

[d]: In order to recognize the other Bridgestone references, PLUM would need to try to treat misspellings, as well as treat the definite reference explicitly.

(2) *Did your system get the OWNERSHIPs, in particular from "... THE REMAINDER BY TAGA CO."?*

PLUM did produce an ownership object with 75 and 15 % ownership percentages; however, the system filled in the owning entities incorrectly. PLUM did not attempt to handle phrases like "the remainder ...". PLUM also missed the capitalization information in this example.

Other comments on walkthrough performance:

The PLUM system found 3 tie-up objects instead of 1. One of the spurious tie-ups resulted from the discourse event triggered by "the new company" not being correctly merged with the earlier mention of the joint venture company. The reason for the second spurious tie-up stems from PLUM having identified "Taiwan" (in the phrase "in Taiwan") as a corporation, and more precisely, as a joint venture company.

The lexical entry for "Taiwan" incorrectly lists it as a corporation, as well as a country. Once "Taiwan" was identified as a corporation, the pattern ["set up" ... <company> with <company>] matched the text "set up a joint venture in Taiwan with a local concern and ...", and "Taiwan" was identified as the joint venture company. Since this joint venture company was found to be different from "Bridgestone Sports Taiwan Co.," which was also identified as a joint venture company by the system, 2 separate tie-ups were generated.

After the test was run, we removed the definition of "Taiwan" as a corporation. With this change, the system generated 1 less tie-up object, and it correctly found the reference between "a joint venture" and "the joint venture." This correction is reflected in the sample event given in the discourse component description section.

"UNION PRECISION CASTING CO" was missed because it was not recognized as a possible company name: capitalization information was not available to help with name recognition (the article was fully capitalized), and the tagging component tagged "casting" as a V, a category which is not allowed to be taken as part of a company name.

Although PLUM did not recognize the typo "Bridgeston Sports", this did not cause any processing problem. It only resulted in PLUM's missing this alias.

PLUM produced 2 activity objects, triggered by the verb "produce" (golf clubs) and the noun "production" (of 20,000 iron and "metal wood" clubs). The first was correct, but the second was spurious.

PLUM recognized some ownership information, including the 75 and 15 percentage shares in the venture. Because PLUM failed to identify Union Precision Casting Co, this entity was not represented in the ownership information. PLUM did not attempt to treat the information conveyed by the phrasing "and the remainder by Taga Co."

EME Walkthrough (Article 2789568)

Questions to Address:
(1) *What information triggers the instantiation of each of the two LITHOGRAPHY objects?*
The PLUM system generated 3 lithography objects, all of type UNKNOWN (the key contains 1 LASER lithography and 1 UNKNOWN lithograpy). The three triggering phrases are: "a new stepper," "the stepper," and "latest stepper"

(2) *What information indicates the role of Nikon Corp. for each Microelectronics Capability?*
The PLUM system initially finds Nikon Corp. as the manufacturer of each of the 3 capabilities (in the key, Nikon is the manufacturer of the LASER lithography and the manufacturer and distributor of the UNKNOWN lithography). Nikon was associated with each of the three capabilities because it occurred in the same sentence. Our statistical model of entity<->capability relationships indicated that Nikon was most likely to be a manufacturer, so it was placed in this role.

We actually found Nikon as a distributor of all 3 capabilities, but we removed this relation as it was determined to be unlikely by our statistical model. Nikon was thought to be a distributor because of the trigger verbs "market" and "sell." The discourse rule then picked up Nikon Corp. (at score 1) as the agent of this verb.

(3) *Explain how your system captured the GRANULARITY information for "The company's latest stepper."*
The granularity phrase "a resolution of 0.45 micron" was correctly understood by the semantics component and was associated with the appropriate lithography object via a discourse rule. However, the granularity filler was ruled out by the template generator because its confidence score fell outside the threshold set for this slot (the threshold setting is tailored to provide the best overall system performance). Consequently, the granularity information did not appear in the response template.

System Response Template:

<TEMPLATE-2789568-1> :=
DOC NR: 2789568
DOC DATE: 191090
DOCUMENT SOURCE: "Comline Electronics"
CONTENT:
<MICROELECTRONICS_CAPABILITY-2789568-1>
<MICROELECTRONICS_CAPABILITY-2789568-2>
<MICROELECTRONICS_CAPABILITY-2789568-3>
<MICROELECTRONICS_CAPABILITY-2789568-1> :=
PROCESS: <LITHOGRAPHY-2789568-1>
MANUFACTURER: <ENTITY-2789568-1>
<MICROELECTRONICS_CAPABILITY-2789568-2> :=
PROCESS: <LITHOGRAPHY-2789568-2>
MANUFACTURER: <ENTITY-2789568-1>
<MICROELECTRONICS_CAPABILITY-2789568-3> :=
PROCESS: <LITHOGRAPHY-2789568-3>
MANUFACTURER: <ENTITY-2789568-1>
<LITHOGRAPHY-2789568-1> :=
TYPE: UNKNOWN
DEVICE: <DEVICE-2789568-1>
EQUIPMENT: <EQUIPMENT-2789568-1>
<LITHOGRAPHY-2789568-2> :=
TYPE: UNKNOWN
EQUIPMENT: <EQUIPMENT-2789568-1>
<LITHOGRAPHY-2789568-3> :=
TYPE: UNKNOWN
EQUIPMENT: <EQUIPMENT-2789568-1>
<ENTITY-2789568-1> :=
NAME: Nikon CORP
TYPE: COMPANY
<DEVICE-2789568-1> :=
FUNCTION: DRAM
<EQUIPMENT-2789568-1> :=
MANUFACTURER: <ENTITY-2789568-1>
MODULES: <EQUIPMENT-2789568-2>
EQUIPMENT_TYPE: STEPPER
STATUS: IN_USE
<EQUIPMENT-2789568-2> :=
MANUFACTURER: <ENTITY-2789568-1>
EQUIPMENT_TYPE: RADIATION_SOURCE
STATUS: IN_USE

(4) *How does your system determine EQUIPMENT_TYPE for "the new stepper"? "the company's latest stepper"?*
Equipment types are defined hierarchically in PLUM's domain model. The word "stepper" is linked to the concept STEPPER in the domain model and triggers a STEPPER discourse event. The template generator translates STEPPER events into equipment objects of type STEPPER. So the equipment_type is based on the domain model concept that is associated with the trigger phrase.

(5) *How does your system determine the STATUS of each equipment object?*
The equipment status is defaulted to IN_USE.

(6) *Why is the DEVICE object only instantiated for LITHOGRPAHY-1?*
PLUM's discourse heuristic for finding a process's device only looks within the same sentence. In this article, the 64-mbit DRAM device is in the same sentence as the first lithography object, but no other.

Other comments on walkthrough performance:
The PLUM system found 3 microelectronics capabilities instead of the 2 in the answer key. The spurious capability results from a discourse referencing problem: the lithography object triggered by the definite phrase "the stepper" was not found to be coreferential with the lithography object triggered by "a new stepper" in the previous sentence. PLUM's definite referencing mechanism is controlled by a parameter When this parameter is turned on, PLUM correctly resolves the definite reference in this example, and only 2 lithography capabilities are generated. However, turning the parameter on negatively affects scores overall, so it was off for the MUC-5 test.

The walkthrough article exemplifies our use of the entity relation statistical model. The PLUM system, through discourse processing, had hypothesized Nikon Corp as both the distributor and manufacturer of <LITHOGRAPHY-2789568-1>, as the distributor of <LITHOGRAPHY-2789568-2>, and as the distributor and purchaser/user of <LITHOGRAPHY-2789568-3>. However, these relations were found at a fairly low confidence by a discourse search rule looking around within the sentence. On the other hand, the statistical model (derived from training data) indicated that Nikon is most likely to be a manufacturer. So the template generator removed the unlikely relations (distributor and purchaser/user) and entered the likely relation of manufacturer. Compared against the key, the statistical model was correct in removing the purchaser/user relation but incorrect in removing one of the distributor relations.

The PLUM system incorrectly generated only 1 stepper equipment object. This was because the discourse event triggered by "the company's latest stepper" was incorrectly merged with the earlier stepper events. If PLUM could have recognized the two granularities and associated them with the 2 different stepper objects (at a high level of confidence), this over-merging error could have been prevented.

The device size information was missed because PLUM failed to correctly analyze the sequence "64- mbit dram." Since running the walkthrough message for the MUC-5 test, this problem has been fixed, so the device size information in this article is now reported.

JJV Walkthrough (Article 0002) appearing on the next page

Questions to Address:

(1) *How does your system determine whether there is a reportable tie-up?*
A tie-up is generated whenever a teikei sentence pattern is matched.

(2) *In Article 0002, how many tie-ups were found? What strategies are used to determine the number of tie-ups in Sentence 2?*
PLUM found 4 tie-ups, the results after merging those which matched sentence patterns.

(3) *How does your system determine the entities in a tie-up?*
Some entities are picked up directly in the semantics when parsed within a fragment, or via lexical clues and syntactic/semantics contexts within phrases. Others are picked up via discourse rules.

(4) *How many discourse entities were identified anywhere in the text, and how did the system determine which of these were reportable?*
The template generator's parameters were set to only output objects directly related to a tie-up.

System Response Template:

```
<テンプレート-0002-1> :=
    記事符号：   0002
    発行年月日：   850108
    ニュース出所：   "朝日新聞   朝刊"
    内容：   <提携-0002-1>
   <提携-0002-2>
   <提携-0002-3>
   <提携-0002-4>
    完了年月日：   930811
<提携-0002-1> :=
    提携状況：   現行
    エンティティー：   <エンティティー-0002-1>
   <エンティティー-0002-2>
<提携-0002-2> :=
    提携状況：   現行
    エンティティー：   <エンティティー-0002-1>
   <エンティティー-0002-2>
<提携-0002-3> :=
    提携状況：   現行
    エンティティー：   <エンティティー-0002-3>
   <エンティティー-0002-4>
<提携-0002-4> :=
    提携状況：   現行
    エンティティー：   <エンティティー-0002-4>
<エンティティー-0002-1> :=
    エンティティー名：   東京海上火災保険
    別名：   "東京海上"
    エンティティー別：   企業
    エンティティー関係：   <エンティティー関係-0002-1>
   <エンティティー関係-0002-2>
<エンティティー-0002-2> :=
    エンティティー名：   大和証券
    エンティティー別：   企業
    エンティティー関係：   <エンティティー関係-0002-1>
   <エンティティー関係-0002-2>
<エンティティー-0002-3> :=
    エンティティー名：   同和火災海上保険
    エンティティー別：   企業
    エンティティー関係：   <エンティティー関係-0002-3>
<エンティティー-0002-4> :=
    エンティティー名：   山一証券
    エンティティー別：   企業
    エンティティー関係：   <エンティティー関係-0002-3>
   <エンティティー関係-0002-4>
<エンティティー関係-0002-1> :=
    エンティティー乙：   <エンティティー-0002-1>
   <エンティティー-0002-2>
    甲対乙関係：   パートナー
    状況：   現在
<エンティティー関係-0002-2> :=
    エンティティー乙：   <エンティティー-0002-1>
   <エンティティー-0002-2>
    甲対乙関係：   パートナー
    状況：   現在
<エンティティー関係-0002-3> :=
    エンティティー乙：   <エンティティー-0002-3>
   <エンティティー-0002-4>
    甲対乙関係：   パートナー
    状況：   現在
<エンティティー関係-0002-4> :=
    エンティティー乙：   <エンティティー-0002-4>
    甲対乙関係：   パートナー
    状況：   現在
```

---

(5) *Explain any difficulties you had in identifying the following:*

a) *the correct number of reportable entities*

Since the system doesn't handle conjunction of company names well, it missed one company.

b) *the correct number of tie-ups (correct, for the sake of this walk-through allows BOTH interpretations described in b) above, even though the key template does not.)*

There was some overgeneration due to under-merging by the discourse component.

c) *the correct links between reportable entities and reportable tie-ups.*

Since entities are only hypothesized through tie-up patterns, this is not a problem.

(6) *How does your system determine aliases for entities?*

The system tests all noun phrases and their parts for concatenations of substrings of a company name.

(7) *What problems were there in detecting the alias for the ENTITY named Toukyou Kaijou Kasai Hoken?*

There was no problem.

(8) *Sentence 2 ends with a general statement about products developed in tie-ups between insurance companies and securities companies. How would your system determine that this is a generic, not a specific reference?*
No tie-up is generated when no proper names of companies are mentioned.

(9) *Discuss any specific analysis your system does to handle terms like "ryousha", which appears in Sentence 2. How does your system deal with the usage of the particle "no" that precedes it?*
Currently, our system does not deal with conjunctions like "ryousha".

JME Walkthrough (Article 000452)

System Response Template for 000452:

```
<テンプレート-000452-1> :=
    記事符号：   000452
    発行年月日：   890804
    ニュース出所：   "日経新聞"
    内容：   <マイクロエレクトロニクス機能-000452-1>
   <マイクロエレクトロニクス機能-000452-2>
    完了年月日：   930811
<マイクロエレクトロニクス機能-000452-1> :=
    方法：   <レイヤリング-000452-1>
    製造者：   <エンティティー-000452-1>
   <エンティティー-000452-2>
    配給者：   <エンティティー-000452-1>
   <エンティティー-000452-2>
<マイクロエレクトロニクス機能-000452-2> :=
    方法：   <レイヤリング-000452-2>
    製造者：   <エンティティー-000452-1>
   <エンティティー-000452-2>
    配給者：   <エンティティー-000452-1>
   <エンティティー-000452-2>
<レイヤリング-000452-1> :=
    膜別：   CVD
    装置：   <装置-000452-1>
<レイヤリング-000452-2> :=
    膜別：   CVD
    装置：   <装置-000452-2>
<エンティティー-000452-1> :=
    エンティティー名：   日本真空技術
    エンティティー別：   企業
<エンティティー-000452-2> :=
    エンティティー名：   ＢＴＵインターナショナル社
    エンティティー別：   企業
<装置-000452-1> :=
    製造者：   <エンティティー-000452-1>
   <エンティティー-000452-2>
    装置別：   CVD装置
    状況：   利用中
<装置-000452-2> :=
    製造者：   <エンティティー-000452-1>
    装置別：   CVD装置
    状況：   利用中
```

Questions to Address:

(1) *How does the system determine the existence of a reportable microelectronics capability?*
If a sentence includes equipment and a verb expressing an ME activity, and it matches a sentence pattern, then a ME capability object is created.

(2) *Three entities are mentioned in this article. How did your system determine which were involved in the ME capability? (If the joint venture company was not selected, was it rejected because its activity was in the future, or some other basis?)*
Only company names which fit the ME capability patterns are considered. Our system did not select the joint venture company, since it did not match any company name patterns.

(3) *How does the system identify company names? How does it associate locations with entities?*
Locations are associated with entities by using patterns like
"...-maker, XXXX (headquarters YYYY ..." and
" ...America's biggest ... company, XXXX"
where XXXX is a company name and YYYY is a location.

(4) *How does your system associate film type with each ME capability? (In this article "CVD" is immediately preceded by "metal film." Will your present strategy allow more remote references?)*
First, film names are extracted by means of clue words. Then, if these names match the sentence patterns, they are matched with film types according to the domain model. The order of <film>, <equipment>, and <verb> is not fixed in the system, but currently must be within the same sentence.

(5) *How does you system determine the existence of reportable equipment? How is equipment type determined? (Would the determination of a new equipment type generate a new ME capability?)*
Equipment names are extracted by means of clue words. Their types are decided according to a hierarchy of equipment types. A new equipment type would not by itself generate a new capability. Equipment objects are only reported if some slot besides STATUS is filled.

ACKNOWLEDGMENTS

The work reported here was supported in part by the Defense Advanced Research Projects Agency and was monitored by the Rome Air Development Center under Contract No. F30602-91-C-0051. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Advanced Research Projects Agency or the United States Government.

REFERENCES

[1] Ayuso, D.M., Boisen, S., Fox, H., Ingria, R., and Weischedel, R. "BBN: Description of the PLUM System as Used for MUC-4", *MUC-4 Proceedings,* 1992.

[2] Iwanska, et.al., "Computational Aspects of Discourse in the Context of MUC-3", *Proceedings of the Third Message Understanding Conference (MUC-3),* 1991.

[3] Matsukawa, T., Miller, S., and Weischedel, R. "Example-Based Correction of Word Segmentation and Part of Speech Labelling", to appear in *Proceedings of the ARPA Workshop on Human Language Technology,* 1993.

[4] Weischedel, R., Ayuso, D.M., Bobrow, R., Boisen, S., Ingria, R., and Palmucci, J., "Partial Parsing, A Report on Work in Progress", *Proceedings of the Fourth ARPA Workshop on Speech and Natural Language,* 1991.

[5] Weischedel, R., Meteer, M., Schwartz, R., Ramshaw, L., and Palmucci, J. "Coping with Ambiguity and Unknown Words through Probabilistic Models", *Computational Linguistics (Special Issue on Using Large Corpora: II)* 19, 359-382, 1993.